融合文本图卷积和集成学习的文本分类方法 *

周玄郎†，邱卫根，张立臣
(广东工业大学 计算机学院，广州 510006)

摘　要：为了提高文本分类的准确率，并解决文本图卷积神经网络对节点特征利用不足的问题，提出了一种新的文本分类模型，其内在融合了文本图卷积和 Stacking 集成学习方法的优点。该模型首先通过文本图卷积神经网络学习文档和词的全局表达以及文档的语法结构信息，再通过集成学习对文本图卷积提取的特征进行二次学习，以弥补文本图卷积节点特征利用不足的问题，提升单标签文本分类的准确率以及整个模型泛化能力。为了降低集成学习的时间消耗，移除了集成学习中的 K 折交叉验证机制。融合算法实现了文本图卷积和 Stacking 集成学习方法的关联，在 R8，R52，MR，Ohsumed，20NG 等数据集上的分类效果相对于传统的分类模型分别提升了 1.5%、2.5%、11%、12%、7%以上，该方法在同领域的分类算法比较中表现优异。

关键词：文本表示；文本分类；文本图卷积；集成学习；融合模型
中图分类号：TP391.1　　doi: 10.19734/j.issn.1001-3695.2022.03.0066

Text classification combining text graph convolution and ensemble learning

Zhou Xuanlang†, Qiu Weigen, Zhang Lichen
(*Faculty of Computer*, *Guangdong University of Technology*, *Guangzhou* 510006, *China*)

Abstract: In order to improve the accuracy of text classification and solve the problem of insufficient utilization of node features by text graph convolution neural network, this paper proposes a new text classification model, which integrates the advantages of text graph convolution and Stacking integrated learning method. The model first learns the global expression of documents and words and the grammatical structure information of documents through text graph convolution neural network, and then secondary learns the features extracted by text graph convolution through integrated learning, so as to make up for the insufficient utilization of text graph convolution node features, and improve the accuracy of single label text classification and the generalization ability of the whole model. In order to reduce the time consumption of ensemble learning, the fusion algorithm removes the k-fold cross verification mechanism in ensemble learning. The fusion algorithm realizes the correlation between text graph convolution and stacking integrated learning method. The classification effect on R8, R52, Mr, Ohsumed, 20ng and other data sets is improved by more than 1.5%, 2.5%, 11%, 12% and 7% respectively compared with the traditional classification model. This method performs well in the comparison of classification algorithms in the same field.

Key words: text representation; text classification; text GCN; ensemble learning; fusion model

0　引言

大数据时代，网络文本数据日益增长，数据量越来越庞大，科学管理和组织这些数据变得尤其重要，由此许多文本处理方法[1]应运而生。文本分类是自然语言处理中非常重要的研究领域之一，大量的应用使用了文本分类技术，例如垃圾邮件检测、新闻过滤、计算表型、观点挖掘、情感分析和文档的组织[1,2]等。

文本分类方法可分为传统方法和深度方法。传统文本分类方法主要采用的是机器学习方法，对文本的表示及分类进行研究。传统的文本特征提取方法，如 n-grams 法，得到文本的表示不够充分，缺少文本的词序关系[2]，这使得文本的表示受到限制，处理方式也不够灵活，且在分类方面，只是采用单个分类器进行分类，分类精度不高。深度学习的文本表示方法，如利用卷积神经网络(CNN)[3]和基于 BiLSTM[4]的循环神经网络(RNN)[5]学习局部连续的单词序列对文本进行表示学习，使文本的表示更加灵活，提升了文本分类的效果。然而这类文本表示方法无法获取句子的语法结构信息以及全局信息，使得分类效果受到限制。另外，CNN 和 RNN 等深度学习模型，受限于欧式结构数据，对于文本这类原本就属于非欧式结构的数据来说，则需要做更多的处理。随着深度学习进一步的发展，图神经网络的研究得到越来越多的关注。研究人员发现，图神经网络非常适合文本这类非欧式结构数据的处理[6]，如文本图卷积模型(Text GCN)[7]，能够在训练中自动学习单词和文档的嵌入。并且图神经网络能够整合文本的结构信息，提升了文本的表征能力。然而，在最终的分类方面，图神经网络模型并没有充分利用神经网络学习到的特征。

为了解决以上问题，并提升文本分类的效果，本文提出了新的文本分类模型 TGCN-S(Text GCN-Stacking)，通过使用 Stacking 集成学习方法，对文本图卷积得到的特征进行拟合训练，解决文本图卷积特征利用不足的问题，提高分类效果和模型的泛化能力；为了提高集成学习的速度，移除了集成学习中的交叉验证机制。该模型的有效性在 R8，R52，MR，Ohsumed 和 20NG 等数据的实验上得到验证。

综上所述，本文提出了新的文本分类模型 TGCN-S(Text GCN-Stacking)，主要贡献和创新点概括如下：

1) 本文利用文本图卷积(Text GCN)获取文本的全局信息和文本的结构信息，解决传统模型无法获取文本的结构信

收稿日期：2022-03-01；**修回日期：**2022-04-19　**基金项目：**国家自然科学基金资助项目(61873068)

作者简介：周玄郎(1996-)，男(通信作者)，江西抚州人，硕士研究生，主要研究方向为自然语言处理(1305291858@qq.com)；邱卫根(1968-)，男，江西临川人，教授，硕导，博士，主要研究方向为人工智能、计算机图形图像学；张立臣(1962-)，男，吉林吉林人，教授，博导，博士，主要研究方向为大数据、信息物理融合系统研究.

息的问题，提升文本的特征表达。

2) 优化 Stacking 集成学习模块，移除 k 折交叉验证，在保证分类效率的同时，降低 Stacking 学习过程的时间消耗。将 softmax 分类器替换为 Stacking 集成学习分类器，有效地解决了文本图卷积特征利用不充分的问题，提升整个模型的分类效果和泛化能力。

3) 融合文本图卷积和集成学习的优点，提出新的文本分类模型——TGCN-S，提高文本分类的准确率。

1　相关工作

1.1　传统的文本分类

传统的文本分类的方法有很多，如支持向量机(SVM)[2]，K 最近邻(KNN)和随机森林(RF)[1]等，这些文本分类方法主要聚焦于文本的表示以及相应算法的研究，例如词袋法和 n-grams 表示法。词袋法将文档划分为一个单词集合，并确定它们在文档中的出现频率。n-grams 法[8]将文本中连续的 n 个词语作为一个对象，再将所有的对象放在一起形成一个集合。词袋法中，文本的最终表示结果与集合中单词顺序无关[2]，这将导致句子语法特性以及单词间的相关性丢失，使得文本表示不够充分，无法的到文本全局信息。相比于词袋法，n-grams 能够的到单词的相关性，但忽略了句子的句法特性，对文本的表示不够充分，且缺乏灵活性，同样的，使得文本的全局信息丢失。

1.2　基于深度学习的文本分类

目前，大多数的文本分类方法是基于深度学习，其中代表性的如应用于语句分类的 CNN[3]，基于双向长短期记忆 BiLSTM[4] 的 RNN，以及 BERT 模型[9]等。

Kim 于 2014 年提出了基于卷积神经网络(CNN)的语句分类[3]，它把一维卷积应用在文本语句上，分类准确度上取得了比较好的结果。Liu，Qiu 等人[5]通过将 LSTM 应用在文本分类中，以学习文本表示，保留文本更长的单词信息，提高了文本的表达能力。Jacob 等人[9]提出了 BERT 模型，一种预训练语言的文本表示模型，在大量文本语料中训练了一个通用的语言表示模，能够捕获单词间更长的依赖。这些模型的出现，很大程度上解决了传统分类方法文本表征不足的问题，但是没有捕获文本的结构信息和全局信息。CNN 与 RNN 都主要是针对局部连续的单词序列，能够很好的捕获文本中的局部信息，但是仍然无法得到语料库中单词的全局共现信息以及文本的结构信息。并且以上模型都局限于欧式结构的数据的学习，对于非欧式结构的数据的处理则会显得捉襟见肘，例如文本数据，如果不进行特殊处理，则很难捕获文本的结构信息。

随着深度学习技术的发展，图神经网络(GNN)的研究得到越来越多的关注。GNN 不仅具有参数共享、降低计算量的优点，而且非常适合文本中单词之间非欧式结构数据的处理，取得了机器学习领域的突破。GNN 还能够提取多尺度的局部空间特征并抽象组合成高层特征。通过图嵌入，GNN 能够学习图的节点、边以及子图的低维度向量表示[8]，突破了一般机器学习需要依赖手工的网络结构设计问题，提高了学习的灵活性。在文献[8]中，Cai 等人证明了图神经网络能够很好的处理具有丰富的关系结构任务，能够在图嵌入的过程中保留图的全局信息。Kipf 和 Welling 等人[10]对图神经网络进行了简化，提出了一种图卷积神经网络模型 GCN，该模型可以捕获高阶邻域特征，提升文本分类的准确率。Yao 等人[7]将 GCN 运用到文本分类中，并提出了 Text GCN 模型，对语料库构建大型的异构图，以句子和单词作为图中的节点，通过 GCN 学习单词和句子嵌入，获取文本中单词的全局信息以及整个文本的结构信息，最后得到文本的特征。

1.3　分类器

目前，不管是传统的文本分类，还是基于深度学习的文本的分类方法，在提取文本的特征后，使用的单一的分类器进行分类，如使用 softmax 得到每个类别的概率，并选择概率最大分类作为文本最终的分类。单一的分类器直接进行分类，使得分类结果一次就确定下来，在出现分类失误的情况下，无法对分类结果进行修正调整。集成学习是由多个弱分类器组成的一个强分类器，可以作为一个整体的分类器用以分类，能够很好的解决单个分类器分类能力不足的问题[11]。集成学习可以分为三类：Boosting 算法和 Bagging 算法以及 Stacking 算法[12]，其中具有代表性的是 Stacking 算法，在灵活性和扩展性方面，Stacking 算法比其他两个算法都要好[13,14]，更具效率优势。Stacking 模型能够灵活高效的对文本进行分类，然而，其分类效果依赖于传入 Stacking 模型的文本特征。

基于以上问题，本文提出了一种融合文本图卷积和 Stacking 集成学习的文本分类方法 TGCN-S，利用文本图卷积提取文本特征，通过集成学习弥补原图卷积特征利用不足的问题，提升文本分类的准确性以及模型的泛化能力。为了降低集成学习的拟合时间，移除了 stacking 集成学习中的交叉验证机制，以提升集成学习部分的拟合速度。

2　本文算法

本文方法通过融合文本图卷积和 Stacking 集成学习方法，提出了一种新的文本分类算法 TGCN-S，该模型结合了文本图卷积和 Stacking 的优点。解决文本图卷积特征利用不足的问题，提高文本分类准确度和模型的泛化能力。为了降低集成学习部分的时间消耗，移除了 Stacking 集成学习中的交叉验证机制，以提升集成学习的拟合速度，提高文本分类的效率。

2.1　TGCN-S 模型结构

本文提出的 (TGCN-S) 模型如下图 1 所示。本文将模型分为特征提取和 Stacking 集成分类两部分。TGCN-S 由 Text GCN 和 Stacking 两个部分连接而成，将 Text GCN 提取的特征作为 Stacking 集成学习的输入，并将 Text GCN 分类结果与 Stacking 第一层分类结果拼接，作为 Stacking 第二层的输入，形成残差连接。这种跳跃式连接的方式提升两个模型之间的关联，增强了 Stacking 第二层输入的特征表达。最终通过 Stacking 的第二部分进行分类，得到文本最后的分类结果。在图 1 中，文本异构图的黑点表示文档，白点表示单词，实线表示文档与单词的联系，虚线表示单词之间的联系，根据文本异构图计算得到的邻接矩阵作为 Text GCN 的输入。

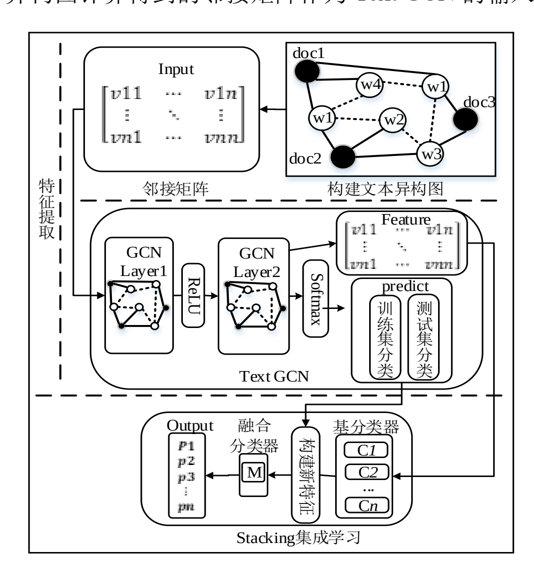


图 1　TGCN-S 总体流程图

Fig. 1　TGCN-S Overall flowchart

2.2 特征提取

本文主要使用 Text GCN 作为特征提取器，作为整个模型的第一部分。在对文本进行构图过程中，将单词和文档作为图的节点，单词与文档之间的连接权值用词频逆文档频率(TF-IDF)表示，单词与单词之间的连接权值使用逐点互信息(PMI)表示。PMI 的计算方式如下：

$$PMI(i,j)=\log\frac{P(i,j)}{P(i)\times P(j)} \tag{1}$$

$$P(i,j)=\frac{N(i,j)}{N} \tag{2}$$

$$P(i)=\frac{N(i)}{N} \tag{3}$$

其中 N 是滑动窗口总数，$N(i,j)$ 表示同时包含节 i , j 的滑动窗口数，$N(i)$ 表示包含节点 i 的滑动窗口数，$P(i,j)$ 表示同时包含节点 i , j 的概率，$P(i)$ 表示滑动窗口包含节点 i 的概率。由此得到节点 i , j 之间的边的权重 A_{ij}，定义如下：

$$A_{ij}=\begin{cases}PMI(i,j) & i,j\ are\ w, PMI(i,j)>0\\ TF-IDF_{ij} & i\ is\ doc, j\ is\ w\\ 1 & i=j\\ 0 & others\end{cases} \tag{4}$$

在式(4)中，w 表示单词，doc 表示一个文档。当 PMI 为正值时，表示语料库中单词的语义相关性较高；当 PMI 为负值时，表示语料库中单词的语义相关性很低或者没有。在构建异构图时，只在 PMI 为正值的节点对直接添加边。之后，再将带权图输入到一个简单的两层 GCN 进行学习。在 GCN 第二层得到词文档嵌入，嵌入的维度与标签类别数大小相同。提取的特征 Z 可以用式(5)计算。最后将节点的嵌入送到 *softmax* 函数中，得到临时的分类输出 Y，如下式(6)计算。

$$Z=\tilde{A}\mathrm{ReLU}(\tilde{A}XW_0)W_1 \tag{5}$$

$$Y=softmax(Z) \tag{6}$$

上述公式中 $\tilde{A}=\tilde{D}^{-\frac{1}{2}}A\tilde{D}^{-\frac{1}{2}}$，而 $\tilde{A}=A+I_N$。A 是 n 阶邻接矩阵，I_N 是 n 阶单位矩阵，n 是顶点个数。$\tilde{D}$ 是 $\tilde{A}$ 对应的度矩阵，其中 $\tilde{D}_{ij}=\sum_j\tilde{A}_{ij}$。$X$ 是由 n 个节点的特征构成的特征矩阵。W_0，W_1 分别是特定于第一层和第二层的可训练的权重矩阵。ReLU 是层间的激活函数。

2.3 集成学习部分

TGCN-S 的第二个部分就是 Stacking 集成学习，传统的 Stacking 集成模型如图 2 所示。

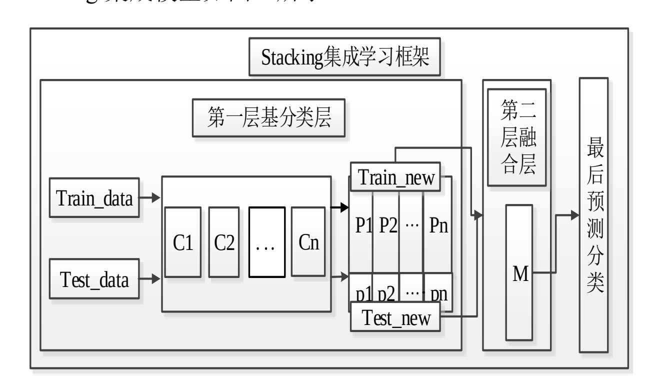


图 2　Stacking 集成学习系统图

Fig. 2　Stacking Diagram of integrated learning systems

传统的 Stacking 集成学习模型(如图 2)对多个基分类器(Ck, (k=1, 2，…，m))进行训练，然后将多个训练好的基分类器对训练集中的数据进行预测得到训练集的预测值 *Pi*(*i*=1, 2，…，m)，再对测试集中的数据进行预测得到测试集对应的预测值 *pj*(j=1, 2, …, m)，最后将多个基分类器得到的预测结果组合在一起，拼接成新数据集，各个基分类器对同一个样本的预测结果组合在一起作为改样本的新特征，训练集得到的预测值组合在一起作为新的训练集特征(*P1, P2, …, Pm*)，测试集得到的预测值组合在一起形成新的测试集特征(*p1, p2, …, pm*)。然后将得到的两组特征集通过 Stacking 第二层融合分类器进行训练和预测，得到最后的分类。

一般地，Text GCN 直接利用 *softmax* 对 GCN 中得到的特征进行分类，并以此作为最终输出，其对训练的特征并没有很好的利用。本文 TGCN-S 模型融合了 Stacking 集成分类以及 Text GCN 优点，在使用 *softmax* 对 GCN 中得到的特征进行分类的过程中，还利用 Stacking 集成学习中各基分类器对 GCN 学习到的特征进行二次拟合，最后进行融合分类，获得文本最终分类结果。

与传统的 Stacking 集成学习不同，TGCN-S 中 Stacking 集成学习部分包含基分类层和融合层。第一层基分类层由 5 个基分类器组成的。第二层融合层除了直接使用各基分类器的分类结果和数据，并整合了 TextGCN 分类的输出结果和数据，即特征提取过程中的训练和预测结果 Y (式(6)所示)，形成跳跃式连接。这种跳跃式连接不仅增强了文本图卷积和 Staking 集成模型之间的联系，而且将 Text GCN 预测效果带入 Stacking 第二层，提升了融合层的分类效果。为了降低集成学习部分的时间消耗，本文去除 Stacking 的交叉验证机制，以提高模型的拟合速度。模型的特征组合过程如图 3 所示，其中 Ci (i=1,2,3,4,5)为基分类器，Tri 为基分类器得到的训练结果，Tei　(i=1,2,3,4,5)为基分类器的预测结果，Train_set 是由各个 Tri 组成的训练集，Test_set 是由各个 Tei 组成的测试集。

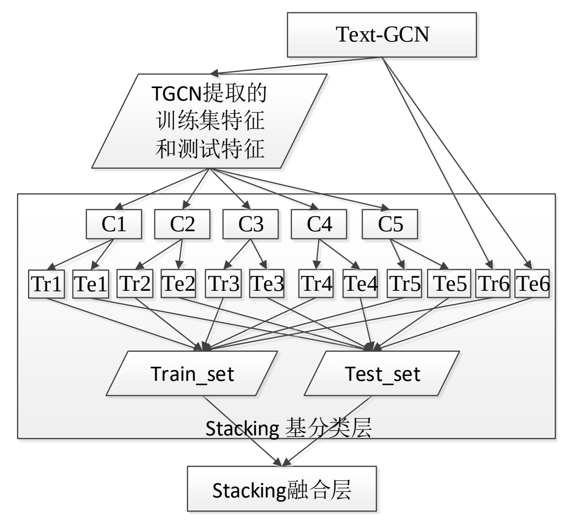


图 3　新特征组合过程示意图

Fig. 3　Schematic of the new feature combination process

Stacking 第一层是由多个基分类器组成，对于基分类器的选择，主要遵循的原则是“各个基分类器准而不同”，不同的基分类器之间要有所差异[12]。本文 Stacking 集成学习的基分类层所采用了 5 种基分类器：支持向量机(SVM)，决策树(DT)，随机森林(RF)，k 最近邻(kNN)以及高斯朴素贝叶斯(Gaussian NB)。一般认为，这 5 种分类器具有基础性的作用，其他大多数分类方法基本上都是基于这 5 个中某一个或多个进行的改进优化。另外，随着模型复杂性和模型数量的增加，模型整体训练的时间必然增加，模型训练拟合开销也会随之增加。基于以上考虑，本文模型 Stacking 第一层的基分类器以上诉 5 种为主。本文中，第二层分类器在单个机器学习分类器预测的基础上，采用投票法(voting)给出最终分类结果。实验结果与分析部分也证明了本文选择的合理性。

3　实验结果和分析

在这个部分，本文通过实验对本文分类模型的分类效果进行验证和分析，并与其他优秀的模型进行对比分析。

3.1 数据集

本文主要使用 R8，Ohsumed，MR，R52 和 20NG 等五种数据集。对所提出的 TGCN-S 模型进行实验对比，分析 TGCN-S 的分类效果。

R8 数据集：R8 数据集分离自路透社语料库，只有 8 个类别，其中有 5485 个训练文档和 2189 个测试文档。

Ohsumed 数据集：是由国家医学图书馆维护的重要的医学文献数目的数据库。提取其中只有单一分类的数据，构成本实验的训练测试用例，其中 3357 个文档用于训练，4043 个文档用于测试，总共 7400 个数据。

MR 数据集：MR 是一个电影评论数据集，每个评论只包含一句话，其中有 5331 篇正面评论，5331 篇负面评论。

R52 数据集：也是分离自路透社语料库，有 52 个类别，有 6532 个训练数据和 2568 个测试数据。

20NG：是一个含有 20 个类别的新闻组数据集，训练集有 11314 个文档，测试集有 7532 个文档。

这些数据由于是文本数据，并不能直接用于模型的训练，因此需要对这些数据集进行预处理[7]。通过预处理，得到表 1 的统计信息，从中可以看到每个数据集训练集和测试集的大小。

表 1 各个数据集的统计信息

Tab. 1 Statistics for each dataset

Dataset	Docs	Training	Test	Words	Nodes	Classes
R8	7674	5485	2189	7688	15362	8
R52	9100	6532	2568	8892	17992	52
Ohsumed	7400	3357	4043	14157	21557	23
MR	10662	7108	3554	18764	29426	2
20NG	18846	11314	7532	42757	61603	20

3.2 对比模型实验数据

本实验部分，本文主要比较的文本分类模型有以下几种：

CNN：针对于文本分类的卷积神经网络[3]，由 Kim 于 2014 年提出，通过在预训练的词向量之上训练的卷积神经网络进行句子级的分类任务。

LSTM：基于长短期记忆文本分类模型，通过使用最后一个隐藏状态作为整个文本的表示形式。由 Liu 等人[5]于 2016 年提出。

Bi-LSTM：双向长短期记忆文本分类模型，是 LSTM 的改版，以预训练的词嵌入作为 Bi-LSTM[4]的输入。

FastText：由 Joulin 等人[15]于 2017 年提出的简单有效文本分类模型，通过将单词 n-gram 嵌入的平均值作为文档的嵌入，再将得到的文档嵌入送入线性分类器进行分类。

Text GCN:文本图卷积[7],由 Yao 等人于 2019 年提出的基于图卷积的文本分类方法，该方法基于单词共现和文档单词关系为整个语料库构建大型异构文本图，再使用图卷积神经网络和 *softmax* 进行学习分类。

通过本文模型与上述几个模型的实验对比，得到不同模型的在不同数据集上的准确率，其各自预测准确率如表 2 所示。

表 2 数据集在各个模型上的预测准确率

Tab. 2 Prediction accuracy of datasets on each model

Model	R8	R52	Ohsumed	MR	20NG
CNN	0.9402	0.8537	0.4397	0.7498	0.7693
LSTM	0.9368	0.8554	0.4113	0.7506	0.6571
Bi-LSTM	0.9631	0.9054	0.4927	0.7768	0.7318
FastText	0.9613	0.9281	0.5770	0.7514	0.7967
TextGCN	0.9707	0.9356	0.6836	0.7674	0.8634
TGCN-S-vote	**0.9858**	**0.9604**	**0.8090**	**0.8828**	**0.9302**

如表 2 所示，本文提出的 TGCN-S 在五个数据集上的测试精度都表现的最好，且有着不同程度的提升。针对 R8 数据集，TGCN-S 的表现比其中最好的 Text GCN 高出了 1.5 个百分点，比其他文本分类算法的精度高出了至少 2 个百分点以上[7]。对于 R52 数据集，本文模型比其他模型高出了 2.5 个百分点以上，相比于 CNN 模型，分类效果提高了 13 个百分点，在 Ohsumed 数据集，TGCN-S 的表现比 Text GCN 模型的表现高出了 12 个百分点，比其他的分类模型高出了 20 个百分点以上[7]。对于 MR 数据集，本文 TGCN-S 模型在测试精度上比 Text GCN 模型高出了接近 11 个百分点[7]，比其他的模型都高出了接近 14 个百分点[7]。在 20NG 这种较大数据集上，TGCN-S 模型也比 TextGCN 模型高出 7 个百分点，比其他模型高出 12 个百分点以上。图 4 直观的展示了各个模型在所用数据集的预测结果。从图 4 中可以看出，本文提出的模型的分类效果都优于对比模型。图上的的数据充分说明，Stacking 集成学习能够对文本图卷积学习到的文本特征进行更高效的利用，能够在不同程度上提升分类效果。

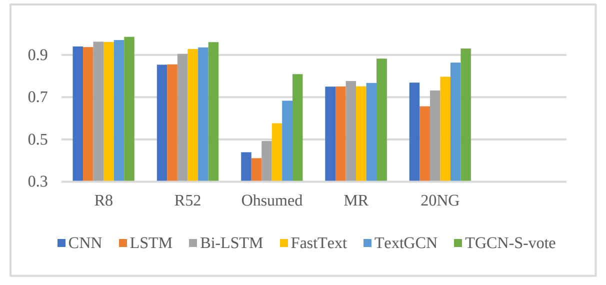


图 4 模型预测结果直方图

Fig. 4 Histogram of model prediction

从表 2 的数据中可以发现，本文提出的模型对不同的数据集的分类效果有着不同的提升，但对于 Ohsumed 和 MR 两个数据集的分类效果没有其他数据集的结果好，原因在于 MR 数据中，存在多个极性评论如"这部电影故事很丰富，但是太恐怖了"。同样的，在 Ohsumed 数据集中各种医学文献之间的描述是相互关联的，在描述某种病例时，会提及与病例有关的药物和信息。图神经网络虽然能捕获文本全局信息，但是无法获取文本内的词序特征，以至于无法提取文本详细的特征，进而导致分类效果欠佳。即便如此，本文方法相对于其他单个分类器来说，仍有非常大的提升。这也说明融合 Stacking 集成学习后的模型，通过投票机制能够有效的提高了文本分类的效果，即便文本中存在多极性的描述，也能得到较高的准确率。这些实验数据也证明本文模型的有效性，可以在很大程度上提升文本分类的准确率。

单一的准确率并不能很好的确定模型的质量，为此，本文采用对比各个模型的宏观 F1(Macro-F1)和微观 F1(Micro-F1)来评估模型的性能。Macro-F1 与 Micro-F1 是综合考虑了模型的查准率和查全率的计算结果，Macro-F1 与 Micro-F1 的值越大说明模型的质量越高，分类性能越好。文献[4]指出，TextGCN 的模型的分类效果和模型质量都优于 CNN，LSTM，BiLSTM，FastText 等模型，因此，本文中主要对 TextGCN 与 TGCN-S-vote 模型的 Macro-F1 与 Micro-F1 值进行比较，以对比判断本文模型 TCGN-S-vote 的性能。各个数据集在两个模型的 Macro-F1 与 Micro-F1 值如下表 3 所示。

表 3 各数据集在 TextGCN 与 TGCN-S-vote 上的 F1 得分

Tab. 3 F1 score for each dataset on textgcn and TGCN-S-vote

评估标准	Model	R8	R52	Ohsumed	MR	20NG
Micro-F1	Text GCN	0.969	0.588	0.674	0.813	0.858
	TGCN-S-vote	**0.977**	**0.831**	**0.781**	**0.876**	**0.930**
Macro-F1	Text GCN	0.933	0.711	0.683	0.758	0.853
	TGCN-S-vote	**0.945**	**0.960**	**0.792**	**0.879**	**0.937**

由表 3 可以看出，本文提出的模型在总体上的 Micro-F1 与 Macro-F1 的得分都比 Text GCN 模型的得分要搞，说明本文提出的模型相比于 TextGCN 模型的质量更高，模型的分类效果也更好。为了对比模型的收敛情况，将 TextGCN 模型与本文提出的模型进行比较，通过每个 epoch 的准确率以及到达稳定时的状态来确定模型的收敛能力，实验结果用折线统计图来表示，如图 5 所示。图 5 分别画出了 MR，R52，R8 数据集在模型 TextGCN 模型和 TGCN-S-vote 模型的各个 epoch 的准确率。从图中可以看到，本文所提出的模型在各个数据集上的收敛速度上都比 TextGCN 要快，都能更早的达到稳定状态。同时，从分类准确率的角度来看，本文提出的模型最终的分类准确率都比 TextGCN 的准确率要高。图 5 的实验数据表明本文模型的有效性。

3.3　去交叉验证的集成学习

为了简化集成学习模块，并提高整个模型的训练预测速度，去除了 Stacking 中所有基分类器的交叉验证机制，只通过随机打乱的方式对训练集和测试集进行处理，并在各个数据集上进行了对比实验。实验结果如表 4 所示。

表 4 去交叉验证对比数据

Tab. 4　Cross-checks the comparison data

Dataset	R8	R52	Ohsumed	MR	20NG
Kfoldt	31.64	225.54	61.02	27.73	253.32
nKfoldt	**11.21**	**76.10**	**33.26**	**5.26**	**100.26**
KP	0.9831	0.9538	0.8357	0.8758	0.9233
nKP	**0.9858**	**0.9604**	**0.8429**	**0.8828**	**0.9334**

在表 4 中，Kfoldt 和 KP 分别表示使用 K 折交叉验证 Stacking 模型所花费的时间及分类准确率，nKfoldt 和 nKP 分别表示未使用 K 折交叉验证 Stacking 部分的耗时及对应的分类准确率。从表 4 中可以发现，不使用 K 折交叉验证的时间消耗低于使用 K 折交叉验证的时间消耗，因为在 Stacking 部分少了 K-1 次的模型的拟合，因此时间有所减少。并且不使用 K 这交叉验证的分类准确率也表现出不低于使用 K 这交叉验证的分类准确率。这是因为，K 折交叉验证原本是在用在数据集较少的情况下，以提高模型的泛化能力，对于数据集较多的情况下，进行交叉验证的效果则收益甚微，还会影响模型的拟合速度，本实验的数据集就是如此。因此，本文提出的模型去除了 Stacking 集成学习交叉验证机制，以降低模型的时间花费，提升模型训练预测的速度的同时保持良好的分类准确率。

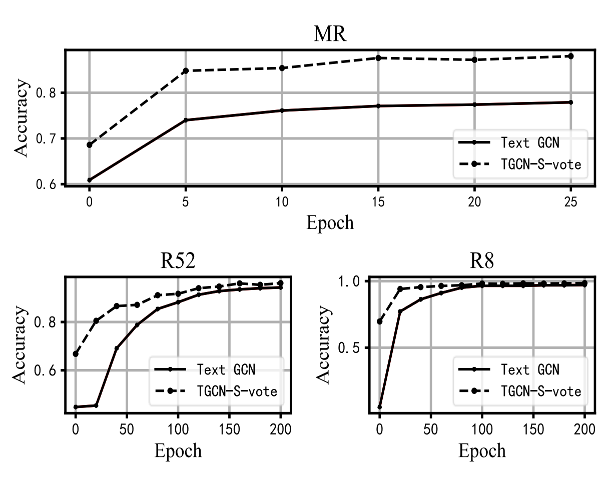


图 5　训练周期与准确率的折线图

Fig. 5　Line chart of training cycle and accuracy

3.4　集成学习融合层的对比实验

为了分析 Stacking 模型中第二层融合分类器 Vote 对最终模型分类效果的影响，本实验通过选择 9 种常用机器学习方法作为 Stacking 第二层的融合分类法，并分别在 R8，R52，Ohsumed，MR 以及 20NG 这五个数据集进行对比实验。九种分类器如下：高斯贝叶斯分类器(GaussianNB),线性回归(LinearRegression), 逻辑回归(LogisticRegression), 决策树(DecisionTree)，LightGBM[16]，支持向量机(SVM)，AdaBoost[17]，Bagging[18]以及 Voting 投票法。实验结果如表 5 所示。

从表 4 中可以看出，使用不同的分类方法作为融合层的分类器会有不同的分类效果。同时，在这五个数据集中，除了在 ohsumed 数据集上，以 LightGBM 作为融合分类器的测试精度略大于投票法之外，其他数据集中，投票法的测试精度都优于其他分类器。这体现了投票法的通用性，且投票法思想简单，易于实现。因此本文提出的模型是以投票法作为 Stacking 模型第二层的融合分类器。

表 5 融合分类器在 R8，R52，Ohsumed，MR，20NG 数据集上的表现

Tab. 5　The performance of each fused classifier on the R8, R52, Ohsumed, MR, 20NG datasets

融合分类器	R8	R52	Ohsumed	MR	20NG
GaussianNB	0.9689	0.9537	0.7057	0.8786	0.7538
LinearRegression	0.9783	0.9537	0.7289	0.7574	0.8192
LogisticRegression	0.9616	0.7736	0.3834	0.8673	0.4067
DecisionTree	0.9726	0.9569	0.7682	0.8632	0.8547
LightGBM	0.9836	0.0901	**0.8217**	0.8584	0.7738
SVM	0.9671	0.8763	0.6747	0.8347	0.7563
AdaBoost	0.9306	0.8000	0.4957	0.8736	0.3604
Bagging	0.9826	0.9532	0.7823	0.8788	0.8320
Vote	**0.9858**	**0.9604**	0.8090	**0.8828**	**0.9302**

4　结束语

本文提出了一种融合文本图卷积(Text GCN)和 Stacking 集成学习的文本分类方法(TGCN-S)，解决 Text GCN 特征利用不足的问题，提高文本分类准确率。不同于传统方法使用单个分类器对文本分类或者深度学习使用 *softmax* 直接对 Text GCN 提取的特征进行分类，TGCN-S 采用 Stacking 集成学习，对 Text GCN 得到的特征进行二次学习，同时，去除 Stacking 集成学习中基分类器的交叉验证机制，加速模型拟合，最后通过融合层得到样本最后的分类。本文 TGCN-S 模型在 R8, R52, MR, Ohsumed 以及 20NG 等数据集上的准确率分别达到了 98.58%，96.04%，88.28%，80.90%，93.02%，相对于其他模型有着很大的提升。实验结果表明本文所提出的模型在文本分类方面具有较高的识别效果，同时也证明了该方法的可行性。

本文对于 Stacking 的基分类器的参数只是凭借经验设置，并没有对这些参数进行优化，未来研究方向可以对这些基分类器的参数进行优化，以进一步提高整个模型的分类效果，提高模型的分类精度。同时图卷积学习到的特征表达缺少语句中的词序关系，因此丰富文本的特征表达也是未来研究方向之一。

参考文献：

[1] Kowsari, Meimandi J, Heidarysafa, *et al.* Text Classification Algorithms: A Survey [J]. Information, 2019, 10 (4): 150.

[2] Li Qian, Peng Hao, Li Jianxin, *et al.* A Survey on Text Classification: From Shallow to Deep Learning [J/OL]. 2020. [2022-04-15]. https://doi. org/10. 48550/arXiv. 2008. 00364.

[3] Kim, Yoon. Convolutional Neural Networks for Sentence Classification [C]. EMNLP. 2014. (2014-09-03) [2022-04-15]. https://doi. org/10. 48550/arXiv. 1408. 5882.

[4] 金宸，李维华，姬晨，等．基于双向 LSTM 神经网络模型的中文分词 [J]．中文信息学报, 2018, 32 (02): 29-37. (Jin Chen, Li Weihua, Ji Chen, *et al.* Bi-directional Long Short-term Memory Neural Networks for Chinese Word Segmentation [J]. Journal of Chinese Information Processing, 2018, 32 (02): 29-37.)

[5] Liu Pengfei, Qiu Xipeng, Huang Xuanjing. Recurrent Neural Network for Text Classification with Multi-Task Learning [J]. Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, 2873-2879.

[6] Zhou Jie, Cui Ganqu, Hu Shengding, *et al.* Graph neural networks: A review of methods and applications [J]. AI Open, 2020, 1: 57-81.

[7] Yao Liang, Mao Chengsheng, Luo Yuan. Graph Convolutional Networks for Text Classification [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33: 7370-7377.

[8] Cai Hongyun, Zheng Vincent W, Chang Chenchuan. A Comprehensive Survey of Graph Embedding: Problems, Techniques and Applications [J]. IEEE Transactions on Knowledge & Data Engineering, 2018, 30 (9): 1616-1637.

[9] Devlin J, Chang M W, Lee K, *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding [J]. arXiv preprint arXiv: 1810. 04805, 2018.

[10] Kipf, T, & Welling, M. Semi-Supervised Classification with Graph Convolutional Networks [C]. ICLR 2017. (2017-02-22) [2022-04-15]. https://doi. org/10. 48550/arXiv. 1609. 02907.

[11] Mehrotra K G, Mohan C K, Huang H, . Ensemble Methods [G]// Terrorism, Security, and Computation. Terrorism, Security, and Computation, 2017: 135–152.

[12] 徐继伟，杨云．集成学习方法：研究综述 [J]．云南大学学报：自然科学版, 2018, 40 (06): 1082-1092. (Xyu Jiwei, Yang Yun. Integrated Learning Methods: Research Review [J]. Journal of Yunnan University: Natural Sciences Edition, 2018, 40 (06): 1082-1092.)

[13] 冉亚鑫，韩红旗，张运良，等．基于 Stacking 集成学习的大规模文本层次分类方法 [J]．情报理论与实践, 2020, 43 (10): 171-176+182. (Ran Yaxin, Han Hongqi, Zhang Yunliang, *et al.* Large-scale Text Hierarchical Classification Method based on Stacking Ensemble Learning [J]. Information Theory and Practice, 2020, 43 (10): 171-176+182.)

[14] 吴挡平，张忠林，曹婷婷．基于 Stacking 策略的稳定性分类器组合模型研究 [J]．小型微型计算机系统, 2019, 40 (05): 135-139. (Wu Dangping, Zhang Zhonglin, Cao Tingting. Research on Stability Classifier Combination Model Based on Stacking Strategy [J]. Small Microcomputer System, 2019, 40 (05): 135-139.)

[15] Joulin A, Grave E, Bojanowski P, *et al.* Bag of Tricks for Efficient Text Classification [C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. 2017.

[16] Ke G, Meng Q, Finley T, *et al.* Lightgbm: A highly efficient gradient boosting decision tree [J]. Advances in neural information processing systems, 2017, 30.

[17] Rehman Javed A, Jalil Z, Atif Moqurrab S, *et al.* Ensemble adaboost classifier for accurate and fast detection of botnet attacks in connected vehicles [J]. Transactions on Emerging Telecommunications Technologies, 2020: e4088.

[18] Wang Qi, Luo Zhihao, Huang Jincai, *et al.* A Novel Ensemble Method for Imbalanced Data Learning: Bagging of Extrapolation-SMOTE SVM [J]. Computational Intelligence and Neuroscience, 2017, 2017: 1827016.